The Journal of Open Source Software

UPOsHam: A Python package for computing unstable periodic orbits in two-degree-of-freedom Hamiltonian systems

Wenyang Lyu[1], **Shibabrat Naik**[1], **and Stephen Wiggins**[1]

1 School of Mathematics, University of Bristol, Fry Building, Woodland Road, Bristol BS8 1UG, United Kingdom

DOI: 10.21105/joss.01684

Software

- Review
- Repository
- Archive

Editor: Kristen Thyng

Reviewers:

- @dpsanders
- @JoshKarpel

Submitted: 21 August 2019
Published: 14 January 2020

Statement of Need

Hamiltonian models are used in a diverse array of systems in natural and engineering sciences, for example, celestial mechanics, ship dynamics, chemical reactions, and structural mechanics. In two-degree-of-freedom Hamiltonian systems, the fundamental phase space structures that partition dynamically distinct trajectories and mediate transitions between multi-stable regions are *stable* and *unstable invariant manifolds* of an *unstable periodic orbit* (UPO). In two-degree-of-freedom systems, the phase space is four dimensional and the dynamics is constrained to the three dimensional energy surface which is partitioned by the two dimensional stable and unstable manifolds of the UPO around an index-1 saddle equilibrium point (see Wiggins, 2016 for more details). Since in this case, the UPO anchors the invariant manifolds that partition trajectories, their computation and stability analysis form the starting point for dynamical systems analysis. UPOsHam is meant to serve this purpose by providing a module of numerical methods, along with example Hamiltonian systems, for computing the unstable periodic orbits at any specified total energy as long as their existence is guaranteed. Even though there are existing numerical methods for computing UPOs, we have found that they either lack in reproducibility, have a steep learning curve for using the software, or have been written using closed source software, and at times combination of these (Farantos, 1998; Pollak, Child, & Pechukas, 1980). Our aim is to provide an open source package that implements some of the standard methods and shows the results in the context of example Hamiltonian systems. This is meant as a starting point to integrate other numerical methods in an open source package such that UPOs computed in dynamical systems papers can be reproduced with minimal tweaking while providing an exploratory environment to further develop the underlying methods.

Summary

The Python package, UPOsHam, is a collection of three methods in the form of submodules under `uposham` for computing UPOs around index-1 saddles in the bottleneck of Hamiltonian systems. When the form is kinetic (purely momenta-dependent terms) plus potential energy, the UPOs project as lines on the configuration space (x, y) and connect opposite points of an equipotential line $V(x, y) = E$. The three methods described below have been implemented as example Hamiltonian systems (also available as submodules under `uposham`) and are described in §:Examples. The demonstration scripts available in the package show how to import each of the methods and implement system-specific functions for computing the UPO.

The computed UPOs using the three methods for the coupled quartic Hamiltonian are compared in Figure 1.

Features: Available Methods

In this package, the user has the option to choose between the three methods described below. These are implemented in separate scripts with functions that can be modified to define the total energy (Hamiltonian), potential energy, vector field, Jacobian, and variational equations (Parker & Chua, 1989).

Turning point

This method is based on finding the UPO by detecting trajectories initialized on the equipotential contour ($V(x, y) = E$ where $V(x, y)$ is the potetial energy function and E is the total energy) that turn in the opposite directions (Pollak et al., 1980). This method relies on the fact that for Hamiltonians of the form kinetic plus potential energy, the UPO is the limiting trajectory that bounces back and forth between the equipotential contour corresponding to the given total energy. To converge on this limiting trajectory, the turning point method iteratively decreases the gap between the bounding trajectories that turn in the opposite directions. Detection of the turning point is done using a dot product condition which leads to stalling of the method beyond a certain tolerance (typically 10^{-6} in the examples here.)

Turning point based on configuration difference

Based on the turning point approach, we have implemented a *new method* which shows stable convergence and does not rely on the dot product formula. Suppose we have found two initial conditions on a given equipotential contour and they turn in the opposite directions. If the difference in x-coordinates is small ($\approx 10^{-2}$), the generated trajectories will approach the UPO from either sides. If the difference in x-coordinates is large, we can integrate the Hamilton's equations for a guess time interval and find the turning point (event using ODE event detection) at which the trajectories bounce back from the far side of the equipotential contour in opposite directions. We choose these two points as our initial guess and the difference of x-coordinates becomes small. Without loss of generality, this method can be modified to either pick the difference of y-coordinates or a combination of x and y coordinates. This choice will depend on the orientation of the potential energy surface's bottleneck in the configuration space.

Differential correction

This method is based on small ($\approx 10^{-5}$) corrections to the initial conditions of an UPO and continuing to desired total energy. The procedure is started from the linear solutions of the Hamilton's equations and which generates a small amplitude ($\approx 10^{-5}$) UPO. This is fed into the procedure that calculates corrections to the initial condition based on errors in the terminal condition of the UPO. This leads to convergence within 3 steps in the sense of the trajectory returning to the initial condition. Once a small amplitude UPO is obtained, numerical continuation increases the amplitude and, correspondingly, the total energy, while a combination of bracketing and bisection method computes the UPO at the desired energy for a specified tolerance (Koon, Lo, Marsden, & Ross, 2011; Naik & Wiggins, 2019).

Example systems

Consider the following two-degree-of-freedom Hamiltonian model where x, y are configuration space coordinates and p_x, p_y are corresponding momenta, $V(x, y)$ is the potential energy, and $T(x, y)$ is the kinetic energy.

Quartic Hamiltonian

This Hamiltonian can be considered as a low dimensional model of a *reaction* in a bath where the coupling is controlled using a parameter. The potential energy is a double-well surface

and the bath is modeled using a harmonic oscillator.

$$\mathcal{H}(x, y, p_x, p_y) = \frac{p_x^2}{2} - \alpha\frac{x^2}{2} + \beta\frac{x^4}{4} + \frac{\omega}{2}\left(p_y^2 + y^2\right) + \frac{\epsilon}{2}(x-y)^2 \tag{1}$$

where $\alpha, \beta, \omega, \epsilon$ are free parameters. When $\epsilon \neq 0$, the system is referred to as the *coupled quartic Hamiltonian*, and *uncoupled quartic Hamiltonian* otherwise.

De Leon-Berne Hamiltonian

This Hamiltonian has been studied as a model of isomerization of a single molecule that undergoes conformational change (De Leon & Berne, 1981; De Leon & Marston, 1989) and exhibits regular and chaotic dynamics relevant for chemical reactions.

$$\mathcal{H}(x, y, p_x, p_y) = T(p_x, p_y) + V_{\mathrm{DB}}(x, y) = \frac{p_x^2}{2m_A} + \frac{p_y^2}{2m_B} + V_{\mathrm{DB}}(x, y) \tag{2}$$

where the potential energy function $V_{\mathrm{DB}}(x, y)$ is

$$\begin{aligned} V_{\mathrm{DB}}(x, y) =& V(x) + V(y) + V(x, y) \\ V(y) =& 4y^2(y^2 - 1) + \epsilon_s \\ V(x) =& D_x\left[1 - \exp(-\lambda x)\right]^2 \\ V(x, y) =& 4y^2(y^2 - 1)\left[\exp(-\zeta\lambda x) - 1\right] \end{aligned} \tag{3}$$

The parameters in the model are m_A, m_B which represent mass of the isomers, while ϵ_s, D_x denote the energy of the saddle, dissociation energy of the Morse oscillator, respectively, and will be kept fixed in this study, λ, ζ denote the range of the Morse oscillator and coupling parameter between the x and y configuration space coordinates, respectively.

Visualization of UPOs

In Fig. 1, we compare the results for the three methods for the coupled quartic Hamiltonian to show that they reproduce each other upto visual inspection.

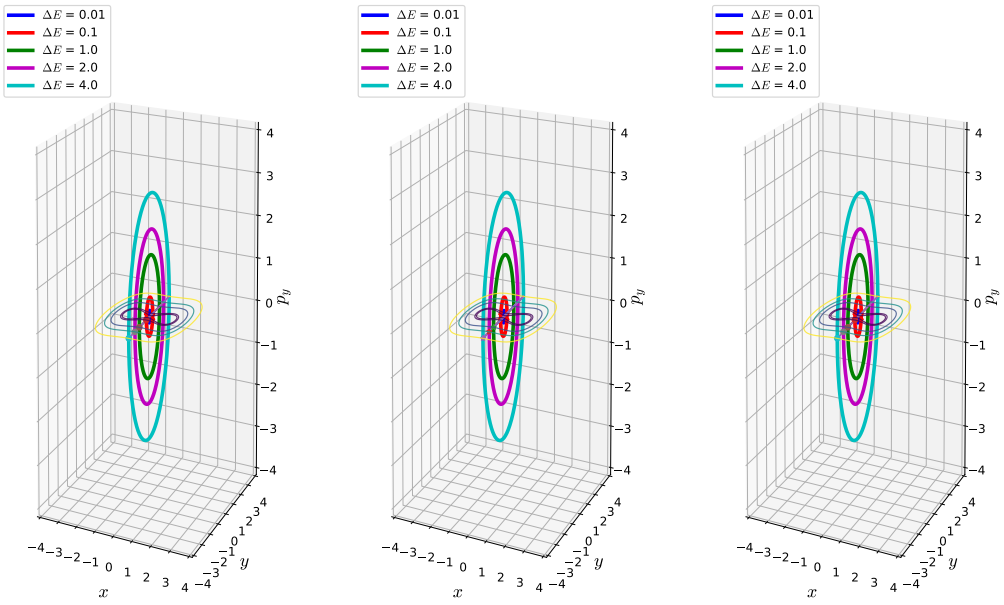

Figure 1: Comparing the UPOs at different total energies computed using the turning point (left), turning point based on configuration difference (center), and differential correction (right) methods for the coupled quartic Hamiltonian. Equipotential contour lines are shown as projection at $p_y = 0$.

In Fig. 2, we compare the results for the *turning point based on configuration difference method* for the three example Hamiltonians and find they are consistent for different total energies.

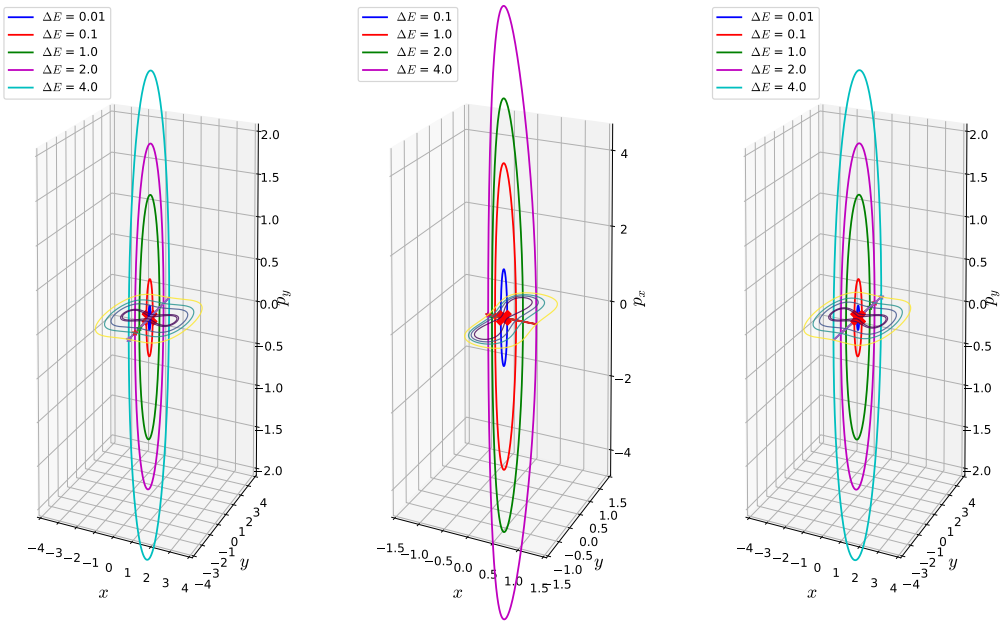

Figure 2: Comparing the UPOs computed using the turning point based on configuration difference method at different total energies for the three example Hamiltonian systems: Uncoupled quartic (left), De Leon-Berne (center), Coupled quartic (right). Equipotential contour lines are shown as projection at $p_y = 0$.

Relation to ongoing research projects

We are developing geometric methods of phase space transport in the context of chemical reaction dynamics that rely on identifying and computing the UPOs. Manuscripts related to the Quartic Hamiltonian and De Leon-Berne Hamiltonian are under preparation.

Acknowledgements

We acknowledge the support of EPSRC Grant No. EP/P021123/1 and Office of Naval Research (Grant No. N00014-01-1-0769). The authors would like to acknowledge the London Mathematical Society and School of Mathematics at the University of Bristol for supporting the undergraduate research bursary 2019. We acknowledge contributions from Shane D. Ross for the early MATLAB version of the differential correction and numerical continuation software package.

References

De Leon, N., & Berne, B. J. (1981). Intramolecular rate process: Isomerization dynamics and the transition to chaos. *The Journal of Chemical Physics*, *75*(7), 3495–3510. doi:10.1063/1.442459

De Leon, N., & Marston, C. C. (1989). Order in chaos and the dynamics and kinetics of unimolecular conformational isomerization. *The Journal of Chemical Physics*, *91*(6), 3405–3425. doi:10.1063/1.456915

Farantos, S. C. (1998). POMULT: A program for computing periodic orbits in hamiltonian systems based on multiple shooting algorithms. *Computer physics communications*, *108*(2-3), 240–258. doi:10.1016/S0010-4655(97)00131-8

Koon, W. S., Lo, M. W., Marsden, J. E., & Ross, S. D. (2011). *Dynamical systems, the three-body problem and space mission design* (p. 327). Marsden books. doi:10.1142/9789812792617_0222

Naik, S., & Wiggins, S. (2019). Finding normally hyperbolic invariant manifolds in two and three degrees of freedom with Hénon-Heiles-type potential. *Phys. Rev. E*, *100*(2), 022204. doi:10.1103/PhysRevE.100.022204

Parker, T. S., & Chua, L. O. (1989). *Practical numerical algorithms for chaotic systems*. New York, NY, USA: Springer-Verlag New York, Inc. doi:10.1007/978-1-4612-3486-9

Pollak, E., Child, M. S., & Pechukas, P. (1980). Classical transition state theory: A lower bound to the reaction probability. *The Journal of Chemical Physics*, *72*(3), 1669–1678. doi:10.1063/1.439276

Wiggins, S. (2016). The role of normally hyperbolic invariant manifolds (NHIMs) in the context of the phase space setting for chemical reaction dynamics. *Regular and Chaotic Dynamics*, *21*(6), 621–638. doi:10.1134/S1560354716060034